IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 17, 2020, accepted July 22, 2020, date of publication July 27, 2020, date of current version August 5, 2020.

*Digital Object Identifier 10.1109/ACCESS.2020.3011989*

# Point Cloud Reduction and Denoising Based on Optimized Downsampling and Bilateral Filtering

**BOCHANG ZOU, HUADONG QIU, AND YUFENG LU**
School of Mechanical and Automotive Engineering, Qilu University of Technology, Jinan 250300, China

Corresponding author: Huadong Qiu (sonntag@126.com)

This work was supported by the Jinan Science and Technology Bureau and undertaken by Qilu University of Technology, Machine Vision-Based Online Intelligent Detection of Foreign Objects in liquids is implemented under Grant 2019GXRC067.

**ABSTRACT** For the issue of using the center of gravity during down-sampling, some points of their feature will be lost. We propose a new method, FWD(Farthest point Weighted mean Down-sampling), this method uses down-sampling to find the center of gravity, it is added to the furthest point sampling and performed ten iterations. The obtained 11-point distance is weighted average to find the feature point. Influences of environmental noise and self-noises on the subsequent processing of point cloud are considered. A PWB (Principal component analysis Wavelet function Bilateral Filtering) method is proposed. The normal vector of points is calculated by PCA. The distance between two points in the optimal neighborhood is obtained by the particle swarm optimization(PSO) method. This method performs wavelet smoothing and utilizes the Gaussian function to retain the edge eigenvalues. FWD simplified 90840 points in 48 seconds in the case of retaining the complete feature points. Compared with other latest methods, better results have been obtained. PWB reached de-noising precision of 0.9696 within 72.31s. Accuracy of de-noising is superior to the latest method. The loss of feature points is completed by FWD, the removal of noise is by PWB. Images of de-noising precision prove the priority of the method. The verification shows that the feature points are retained and the noise is eliminated.

**INDEX TERMS** Furthest point sampling, down-sampling, principal component analysis (PCA), wavelet function, bilateral filtering, particle swarm optimization (PSO).

## I. INTRODUCTION

In machine vision applications, point cloud acquisition can contain excessive noise due to the surrounding environment, equipment, lighting, etc. Regarding the removal of noise, the latest research is mainly divided into traditional methods and methods of artificial neural network.

Researches based on artificial neural networks have made considerable progress, a batch normalization de-noising network was designed by Tian *et al.* [1], the problems of covariant displacement and small batches are solved, but the method is only suitable for high-light images, it is not suitable for low-light and blurred images. The separable noise orthogonal transform feature was introduced into the neural de-noising framework by Shin *et al.* [2]. Compared with single-input networks, multi-input networks have good de-noising characteristics. CNN de-noising framework based on residual learning was proposed by Shi *et al.* [3]. The problem of super-resolution de-noising is solved. But the processing effect of Gaussian noise is not a good. In other aspects, the framework of neural network has also been enhanced. Four multi-layer network embedding algorithms based on non-negative matrix factorization (NMF) were proposed by Lu *et al.* [4], clustering and accuracy are improved, but the dimension of space needs to be predefined. In other methods, the appropriate Lyapunov-Krasovsky function is used to optimize the feedback neural network [5], the coefficient of this method can be further optimized. The updated algorithm of the framework also includes sun's [6] feedback controller based on sliding mode control. Two quantized control schemes are proposed with uniform quantizer and logarithmic quantization by Sun *et al.* [7]. There are other studies [8]–[12]. In a word, these methods have achieved certain results, but there is still room for improvement in blurred images, Gaussian noise removal, frame optimization, etc.

The traditional method has also made a lot of progressing. The problem of pixel intensity values being

The associate editor coordinating the review of this manuscript and approving it for publication was Shiping Wen.

contaminated by different types of noise was solved by Elhoseny and Shankar [13]. He proposed the optimal BF model and MI classification denoising approach, which removes excess noise but is computationally intensive. The conventional iterative denoising method also takes a long time. Li *et al.* [14] used an adaptive bilateral filter that retains the edge eigenvalues. However, it takes a long time to perform the adaptive configuration and may lose parts of the point cloud data when scaling the model. The point cloud image is affected by noise in the geometric space and color space. Chinthaka *et al.* [15] achieved ideal denoising characteristics through GLR and GTV point cloud processing, but the execution time is relatively slow. Because point cloud data obtained from external light sources inevitably contains noise, Zeng *et al.* [16] used the prior of the patch flow for denoising, which retains better edge characteristics. However, the method only achieves good results when the image is low-pass filtered and, in other situations, ideal denoising is not achieved.

In this research, aiming at the defects of environmental noise and existing de-noising methods, a method combining FWD and PWB was proposed by us. We have considered the optimization of parameters in the method. The best de-noising effect is guaranteed. In order to optimize the coefficients, new calculation methods are introduced. The result of sampling is judged by using the amount of noise removal. The de-noising effect is compared in the experimental model and the benchmark data by using the de-noising accuracy($P_d$), de-noising recall rate($R_d$), peak signal to noise ratio(PSNR), structural similarity index(SSIM), Information Fidelity Criterion(IFC). The results are compared with different latest methods.

Evaluation parameters have achieved higher results, more accurate than other methods. This method is superior to other existing methods based on artificial neural networks and traditional methods.

## II. POINT CLOUD PROCESSING

### A. POINT CLOUD ACQUISITION

In this paper, we obtain point cloud information about an object and environment under test conditions using a Tuyang FMB10 1X depth camera. The initial image used in the experiment was obtained through the Point Cloud Learning platform in C++ based on Visual Studio 2015.

### B. POINT CLOUD PREPROCESSING

There are many measurement errors in the point cloud obtained for this article. For example, there are many outliers, zigzag convex points, and holes on the surface of the point cloud. Image segmentation will be hindered due to excessive point cloud data; therefore, the preprocessing used in this paper involves point cloud reduction and denoising.

#### 1) SIMPLIFICATION OF A POINT CLOUD

The point cloud acquired by the three-dimensional imaging device is divided into a chaotic point cloud, linear point cloud, and geometric point cloud. The point cloud is geometric and cluttered by the depth camera. The burden is increased by the huge amount of point cloud data. A series of algorithms [17]–[22] is adopted without reducing the feature data. Noise is recognized and feature data is retained. For streamlined three-dimensional (3D) point clouds, optimized down-sampling and geometric sampling have been used in recent studies. However, a single downsampling cannot be used to ideally streamline the point cloud in the bump feature. A 3D voxel grid is constructed and the center of gravity obtained. Some key points are easily missed; for example, those points have key features on the 3D voxel grid. The principle of point cloud curvature is adopted by single geometric sampling and eligible point clouds are filtered by setting angle thresholds. It cannot be applied to the selection of point clouds in a plane. Therefore, we propose a new sampling method that is suitable for both planar and uneven situations.

A simplified algorithm FWD (Farthest Point Weighted Mean Down-sampling) is proposed based on optimized down-sampling, and single sampling is improved on. After constructing the 3D voxel grid, we find the center of gravity. The farthest-point sampling method in uniform sampling is combined with down-sampling and is then weighted. The processing steps are as follows:

(1) A 3D voxel grid is constructed, and the center of gravity O of all points is found in the grid.

(2) The farthest sampling point is used in uniform sampling and the center of gravity O is used as the initial point. We look for the point $\boldsymbol{O_1}$ on the 3D voxel grid that is farthest from O. Let us take this point $\boldsymbol{O_1}$ as the initial point. The farthest point $\boldsymbol{O_2}$ is found in the 3D voxel grid except O, we proceed in turn until we obtain $\boldsymbol{O_{10}}$. Because characteristics and efficiency are guaranteed at the same time. When fewer points are taken, the faster the speed. Therefore, this article temporarily takes 1∼20 points to conduct experiments separately. The result is shown in Fig 1.

(3) Eleven points (O and $O_1 - O_{10}$) are weighted and averaged to obtain the weighted average d, which is taken as the voxel streamlined feature point; that is, the desired point.

$$d = \frac{b_1 a_{1+\cdots+} b_n a_n}{10} \tag{1}$$

where d represents the weighted average, that is the voxel point sought. The distance value from the starting point to the furthest point is represented by $a_n$ (where n is 1–10, and $n \leq 10$). The number of the same $a_n$ is represented by $b_n$.

#### 2) POINT CLOUD DE-NOISING

Low- and high-frequency noise exist in the initial point cloud obtained in the experiment. High-frequency noise can be divided into marginal outliers, isolated outliers, and biased outliers. The next point cloud processing stage will be hindered by these point clouds. Therefore, point cloud denoising

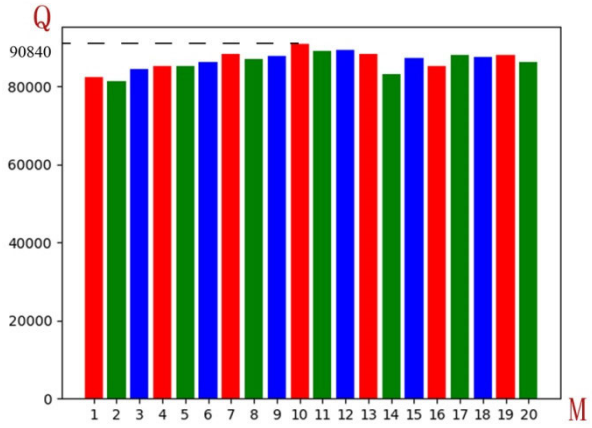



**FIGURE 1.** Here, Q represents the number of points removed and M represents the number of points selected. Because the amount of calculation needs to be minimized, the value of M cannot be too large, so use values of 1 to 20 for the follow-up test. When 10 is used, the number of removals is 90,840, which is larger than in other cases.

needs to be carried out. In the latest de-noising scheme, large-scale outliers are removed by the improved c-means method (FCM) [23]. High-frequency noise is reduced by bilateral filtering. The outliers are judged by setting thresholds in the plan. Good results were achieved. However, the method assumes that the point cloud follows a Gaussian distribution, and other circumstances are not considered. Residual noise is caused. The point cloud denoising methods of the optimized PCA and bilateral filter are used by us.

### *a: AN OPTIMIZED PCA METHOD*

A vertex P of the point cloud is taken and k-means sampling [24] is used. A threshold is set to 3, according to an experimental analysis of the threshold value (Fig 2). The value principle is based on the amount of de-noising, where more gives a better effect. A threshold of 3 gives the best effect. When the point is less than or equal to this threshold in the field, P is an isolated outlier or an edge outlier and can be removed directly from the point cloud. As shown in Fig 3, A is an isolated outlier, while B and C are marginal outliers. When the point in the field is $>3$, this shows that the dot is adjacent to most pixels. Valid feature points are judged and will not be removed directly; for example, D.

The outliers are moved by the optimized principal component analysis (PCA) method of Zhang *et al.* [25]. The 3D field is reduced to 2D through PCA. The principle of 3D reduction to 2D is to project the 3D field onto a 2D plane according to certain rules to obtain the tangent plane of the point cloud. The subsequent steps are as follows:

(1) The y-direction of the coordinate system where the point cloud to be measured is found, and the tangent plane parallel to the y-direction is taken. The normal vector of this plane is taken as the normal of the point cloud.

(2) Let a point on the tangent plane be p, and $p_i$ be the k-nearest neighbor of p. Then, a covariance matrix can be derived and solved.

$$\mathrm{Cov} = \frac{1}{m}\begin{bmatrix} p_i - p_1 \\ p_i - p_2 \\ \cdots \\ p_i - p_k \end{bmatrix}\begin{bmatrix} p_i - p_1 \\ p_i - p_2 \\ \cdots \\ p_i - p_k \end{bmatrix}^T \quad (2)$$

(3) The minimum value (n) of the eigenvalue and eigenvector is taken as the normal vector of the tangent plane.

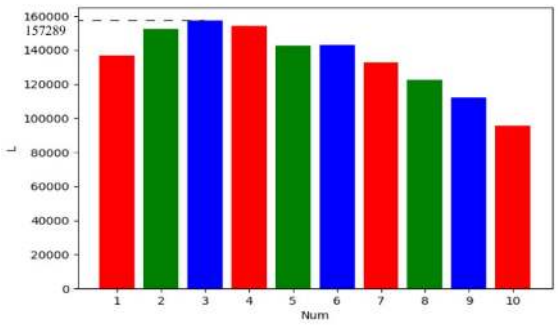



**FIGURE 2.** Here, L represents the amount of noise removed and Num represents the size of the threshold. When the threshold is too high, noise points will be mistaken for feature points. Therefore, the threshold cannot be too large. We tried values of 1–10.

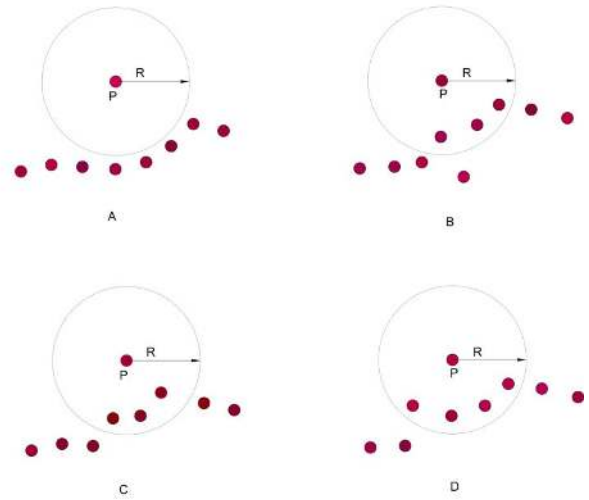



**FIGURE 3.** Here, A is an isolated outlier, B and C are two different marginal outliers, D is the effective feature point.

### *b: A BILATERAL FILTERING ALGORITHM OPTIMIZED WITH WAVELET FILTERING*

In the filtering process, the pixel relationships in the definition domain are not considered by the latest Gaussian filtering and mean filtering processes. Image blur is caused at the edges of pixels. Filtering in the spatial domain and definition domain can be achieved by bilateral filtering at the same time to achieve edge protection, noise reduction, and smoothing. The weighted average method is adopted by the basic bilateral filtering. The pixel values in the area of a point are weighted and averaged as a pixel, which will cause blurring of the edges. The parameters have to be optimized in the method.

There are various latest optimization algorithms. A novel learning rate method was proposed by Ren *et al.* [26], the convergence speed of EBP is improved, but the learning rate still has room for improvement. The evolutionary extreme learning machine is optimized by Niu *et al.* [27], the parameters are optimized by the gravity search algorithm, the ideal result is achieved, but the calculation result of VMD may be biased. A new PSO based on evolutionary extreme learning machine was proposed by Feng *et al.* [28]. The algorithm is optimized. It has achieved satisfactory results in forecasting, but there is the problem of premature convergence. The PSO algorithm has achieved certain results. In research, PSO is utilized to optimize bilateral filtering.

A new PSO wavelets bilateral filtering (PWB) method is proposed, which is based on PSO and a bilateral filter optimized by wavelets. Bilateral filtering is a process that raises the point coordinates. It is added to the fourth vector (pixel value) based on the coordinates. A point $A_i$ is selected in the spatial domain R, and correspondingly expressed as $(i_{A_i}, j_{A_i}, z_{A_i}, I_{A_i})$, where $I_A$ represents the pixel value of A. The weight coefficient $\alpha$ is optimized by pixel processing. This achieves noise removal and feature retention.

The point $A_i$ is selected in the point cloud with position coordinates $(i_{A_i}, j_{A_i}, z_{A_i})$. The selected points are added to the pixel value $(i_{A_i}, j_{A_i}, z_{A_i}, I_{A_i})$ in the space and the distance between two pixels is measured by the Euclidean distance. Another point is that $A_j$ $(i_{A_j}, j_{A_j}, z_{A_j}, I_{A_j})$ is taken in the k neighborhood according to the Equation:

$$\begin{aligned}&\left\|A_j - A_i\right\| \\ &= \sqrt{\left(i_{A_j} - i_{A_i}\right)^2 + \left(j_{A_j} - j_{A_i}\right)^2 + \left(z_{A_j} - z_{A_i}\right)^2 + \left(I_{A_j} - I_{A_i}\right)^2}\end{aligned} \tag{3}$$

Assume that the difference in coordinate values is constant; when its pixel difference is larger, its d value is larger. The magnitude of the influence is judged by the distance. This has a big influence when the difference is larger, making the point clouds blur easily; on the contrary, the impact is small.

The most suitable points are obtained by optimized PSO [29]. Inertia coefficient is improved by adopting nonlinear weighting strategy [30]. The PSO is used to find the point that minimizes $\left(I_{A_j} - I_{A_i}\right)$, And the point cannot be $A_i$.

The principle of PSO refers to Equation 4,5, the current position and speed of this point are $x_i$ and $v_i$. The optimal position P is found, the global optimal position is $P_{id,t}$, the local optimal position is $P_{gd,t}$. The position and velocity of each generation of particles are z and m, respectively.

$$x_{id,t+1} = x_{id,t} + \beta \cdot v_{id,t+1}, \quad 1 \le i, d \le n \tag{4}$$

$$\begin{aligned}v_{id,t+1} = {} & \varphi \cdot v_{id,t} + c_1 \cdot r_1 \cdot \left[P_{id,t} - x_{id,t}\right] \\ & + c_2 \cdot r_2 \cdot \left[P_{gd,t} - x_{id,t}\right], \quad 1 \le i, d \le n\end{aligned} \tag{5}$$

where, the weighting factor is $\varphi$, when it is large, the ability of global search is strong, when it is small, the ability of local search is strong. $c_1$ and $c_2$ are non-negative, they are acceleration constants. $r_1$ and $r_2$ are random numbers in the range of [0, 1]. $\beta$ is a limiting parameter which constrains the flight speed of particles. The inertia coefficient is optimized by the non-linear weighting strategy, it is shown in Equation 6:

$$\begin{aligned}\varphi = {} & (\varphi_{max} - \varphi_{min}) \cdot \left(\frac{i_d}{M}\right)^3 + (\varphi_{max} - \varphi_{min}) \cdot \left(\frac{i_d}{M}\right)^2 \\ & + (\varphi_{max} - \varphi_{min}) \cdot \left(\frac{i_d}{M}\right)\end{aligned} \tag{6}$$

In Equation 6: The maximum and minimum weights are $\varphi_{max}$ and $\varphi_{min}$ respectively. The number of iterations is $i_d$. The total number of iteration steps is M. The validity of the algorithm has been verified by Ren [30].

Optimal value of $\left(I_{A_j} - I_{A_i}\right)$ is obtained by PSO. The value of $\left(I_{A_j} - I_{A_i}\right)$ is guaranteed to decrease and the influence of domain pixels is improved upon filtering.

Bilateral filtering is performed according to the following formula:

$$A' = A_i + \alpha N \tag{7}$$

where $A'$ are the points that obtained through optimized bilateral filtering, $A_i$ is a point in the original point cloud, $\alpha$ is the weighting factor, N is the normal vector after taking the weighted average of pixels in the domain of A. N is parallel to the plane obtained by PCA. The acquired point is moved in the defined direction. This method can avoid the loss of points better than the normal vector movement approach of Zhang *et al.* [25]. The experimental analysis of this comparison result is in the fourth chapter of this article. The filtering weight coefficient is $\alpha$, which is calculated as:

$$\alpha = \frac{\sum_{A_j \epsilon L(A_i)} F_C\left(\left\|A_j - A_i\right\|\right) F_S\left(\left\|\langle n_i - n_j\rangle - 1\right\|\right) \langle A_j - A_i, n_j\rangle}{\sum_{A_j \epsilon L(A_i)} F_C\left(\left\|A_j - A_i\right\|\right) F_S\left(\left\|\langle n_i - n_j\rangle - 1\right\|\right)} \tag{8}$$

where $L(A_i)$ is the immediate point of $A_i$ and $A_j$ is a point in $L(A_i)$. The distance between two points is represented by $\left\|A_j - A_i\right\|$. The inner product of two normal vectors is represent by $\langle n_i - n_j\rangle$, $n_i$ is the normal vector of $A_i$, $n_j$ is the normal vector of $A_j$. $F_c$ is the smoothing weighting factor of point $A_i$ in relation to its immediate neighbors. $F_S$ is the feature reservation weight of point cloud $A_i$ in the normal vector $n_i$ factor for its domain point. When the pixels at the two points differ greatly, the weight function coefficient is small. Point cloud smoothing is convolved by wavelet filtering [31], as shown in Equation 9:

$$F_{C_1}\left(i_{A_i}, j_{A_i}, z_{A_i}\right) = F_{C_0}\left(i_{A_i}, j_{A_i}, z_{A_i}\right) G\left(i_{A_i}, j_{A_i}, z_{A_i}, \sigma_c\right) \tag{9}$$

$$\begin{aligned}G\left(i_{A_i}, j_{A_i}, z_{A_i}, \sigma_c\right) = {} & \frac{1}{\sqrt{2\pi\sigma_c}^3}\left(\frac{\partial^2}{\partial i_{A_i}^2} e^{-\frac{i_{A_i}^2 + j_{A_i}^2 + z_{A_i}^2}{2\sigma_c^2}}\right. \\ & + \frac{\partial^2}{\partial j_{A_i}^2} e^{-\frac{i_{A_i}^2 + j_{A_i}^2 + z_{A_i}^2}{2\sigma_c^2}} \\ & \left. + \frac{\partial^2}{\partial z_{A_i}^2} e^{-\frac{i_{A_i}^2 + j_{A_i}^2 + z_{A_i}^2}{2\sigma_c^2}}\right)\end{aligned} \tag{10}$$

where $\sigma_c$ is the standard variance of the Gaussian function. It is the scale of the wavelet function. The coordinate values of point cloud $A_i$ are $(i_{A_i}, j_{A_i}, z_{A_i})$. The smoothing effect can be changed by varying the magnitude of $\sigma_c$. $F_{C_1}$ can be obtained by finite window convolution; we set the window at 5*5*5.

$$F_{C_1}(i_{A_i}, j_{A_i}, z_{A_i}) = \sum_{k=-2}^{2} \sum_{m=-2}^{2} \sum_{h=-2}^{2} F_{C_0}(i_{A_i} + k, j_{A_i} + m, z_{A_i} + h)\, G(k, m, h, \sigma_c) \tag{11}$$

The variance of the Gaussian function is $\sigma_c$. When the value of $\sigma_c$ is larger, the amplitude of the Gaussian function is expanded in the time domain and the frequency i is reduced in the frequency domain, the greater the selectivity of the field of point $A_i$. The smoothing effect of the point cloud is determined by the size of $\sigma_c$. We take a $\sigma_c$ value for convolution to obtain the image smoothing effect. If the expected result is not reached, another value of $\sigma_c$ is taken. When the result reaches the expected value, the experiment is stopped.

A one-dimensional Gaussian function is adopted as the retention function of edge features.

$$F_S = e^{-\frac{z_{A_i}^2}{2\sigma_S^2}} \tag{12}$$

The distance vector from point $A_i$ to the domain point is $d_1$, and d1 is projected as $d_2$ in the direction of the normal vector of the field point $A_i$. The influencing factor of $d_2$ is $\sigma_S$.

The entire process of the optimized bilateral filtering algorithm is shown in the following seven steps:

(1) A point $A_i$ is taken in the point cloud.
(2) The k neighborhood of $A_i$ is obtained, a point $A_j$ is taken in the field.
(3) The optimal value of $\|A_j - A_i\|$ is found. Optimal values are obtained through the optimized PSO.
(4) The pixel's normal vector N is obtained by the optimized PCA algorithm.
(5) Equation 11 is calculated, the value of $\sigma_c$ is selected for convolution, and the optimal smoothing parameter $F_C$ is obtained.
(6) Equation 12 is adopted to retain the edge features, which conforms to the standard one-dimensional Gaussian function. The $\sigma_S$ value is selected and the $F_S$ value is derived.
(7) Filtering result $A'$ is calculated, $A' = A_i + \alpha N$.

## III. EXPERIMENT AND ANALYSIS OF RESULTS

### A. DATASETS

In this paper, we obtain point cloud information about an object and environment under test conditions using a Tuyang FMB10 1X depth camera. The datasets for the verification of the effect on the basic data sets are Stanford University's scan dataset, RGB-D dataset, SET5, SET14 and BSDS100. Fig 4 presents some examples of partial datasets.

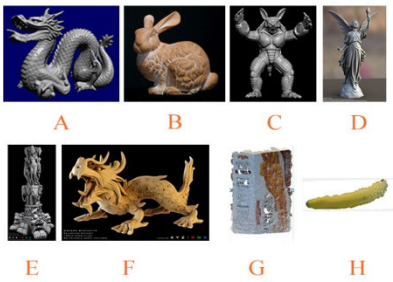



**FIGURE 4.** Here, A, B, C, D, E, F are some examples of Stanford University scan data sets. G and H are two examples of RGB-D data sets.

### B. POINT CLOUD STREAMLINING

The original point cloud was collected by C++ based on Visual Studio 2015. The point cloud was sampled by the FWD method on Matlab and C++ platforms. Feature points were optimized and retained by a weighted average. The original and simplified point cloud diagram obtained by the FWD method is shown in Fig 5.

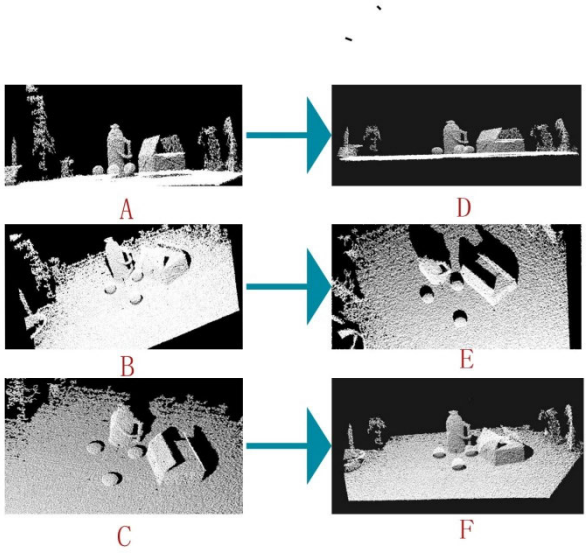



**FIGURE 5.** Point cloud image processed by FWUS.

The point cloud data is obtained three times, shown as Fig 5 A, B, and C. The outline of the object can be roughly recognized by the point cloud in the figure. The acquired point cloud is affected by factors from the outside world and the camera itself. For example, the image acquired is seriously affected by noise due to excessive light and dense point clouds in B. The following effects are obtained using the FWD method. In Fig 5, panel D is the processed result of panel A, while panel E is the processed result of panel B, and panel F is the processed result of panel C. The points around the feature points indicating the same features are removed by the FWD method. The most representative point cloud is found by a combination of the 3D voxel grid and the furthest point sampling.

In a short period of time, the three models reduced the point cloud dataset by 90%, as shown in Table 1. The streamlining of point clouds mainly depends on the accuracy with which feature points are retained. While points representing shape features are preserved, extra points can be deleted. The data in Table 1 corresponds to the processing results of Fig 5. The basic characteristics of the object are preserved. Another criterion for streamlining is timeliness [32]. It can be seen from Table 1 that the FWD method takes less time to eliminate a lot of points.

**TABLE 1. Basic information of point cloud processing.**

| Point cloud | Number of initial points before processing | Number of points after down-sampling treatment | Runtime |
|---|---|---|---|
| A | 342,107 | 41,386 | 98.15 s |
| B | 332,025 | 37,697 | 96.42 s |
| C | 341,125 | 39,442 | 97.51 s |

In Fig 6, the original point cloud obtained is A, while the point cloud obtained by processing of the original with down-sampling is B. Point cloud C was obtained by the fractal algorithm, while point cloud D was obtained through the FWD method. It can be seen from Fig 6 that the results of the FWD method are more simplified. Experiments show that under the same model, the FWD method is more streamlined and better than the latest fractal method. When point cloud models are the same, the FWD algorithm is significantly better than the fractal algorithm and the original downsampling method in terms of the final number of removed points. From Table 2, the FWD method is significantly better than the other two methods in terms of the number of point clouds removed per minute. Good results have been achieved in the simplification of the uneven sphere point cloud. This solves the problem of it being difficult to achieve optimal simplification on an uneven surface and a plane at the same time.

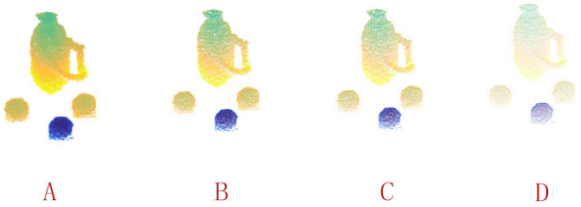



**FIGURE 6. Here, the original data is represented by A. The result of the original down-sampling process is represented by B. The result of Fractal bubble algorithm is represented by C. The result of FWD processing is represented by D.**

The research method has achieved ideal results on the experimental datasets. In the next work, we examine the performance of FWD on the benchmark dataset. It is shown in Fig. 7

**TABLE 2. Basic information of point cloud processing.**

| Algorithm | Number of initial points before processing | Number of points after down-sampling treatment | Runtime (s) | Number of points removed per second |
|---|---|---|---|---|
| Basic down-sampling | 100,865 | 47,895 | 298 | 177 |
| Fractal bubble algorithm [33] | 100,865 | 24,671 | 134 | 569 |
| FWD | 100,865 | 10,025 | 48 | 1893 |

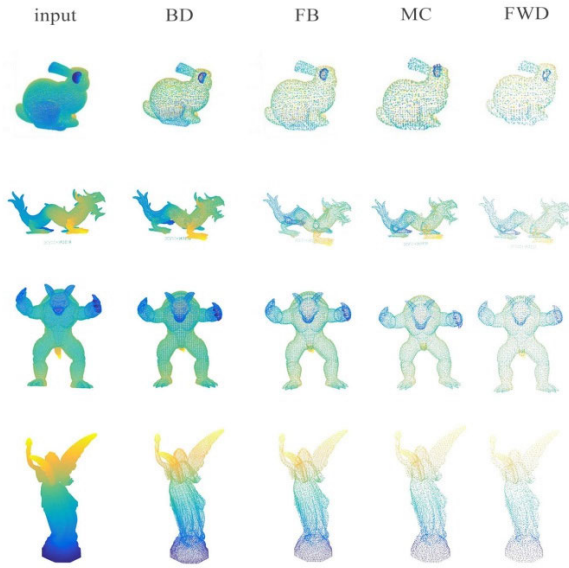



**FIGURE 7. Here, the column where 'input' is represented the original image of the input. The column where'BD' is located represents the image processed by basic down-sampling. The column where'FB' is located represents the image processed by algorithm of fractal bubble. The column where 'MC' is located represents the image processed by MC [22]. The column where'FWD' is located represents the image processed by FWD.**

Fig 7 shows the simplified processing of point cloud images by four methods of latest and different. It can be seen from the results that the four methods can clearly see the shape characteristics of the point cloud. The characteristics of the point cloud are kept intact. In the case where the elementary point cloud shape features are saved, the most sparse point cloud image is obtained by FWD. It holds the smallest number of point clouds. Because the number of point clouds is small, it can reduce the computational burden for subsequent processing.

It can be seen from the results of the experimental model and basic data set that the FWD proposed by the study is

superior to the latest other methods. FWD has achieved ideal results.

### C. DE-NOISING OF A POINT CLOUD

The parameters are optimized by the improved PSO in this study. The most suitable value is obtained through the set principle. After different iterations, different de-noising effects are obtained separately. When the target value is achieved, the iteration stops. The latest and better method for optimization parameters is proposed by Ren. Now, we compare the PWB method with it. The effect of PWB is explored. Our research takes the correlation between the de-noising accuracy and number of iterations as evaluation criteria, it is shown in Fig 8.

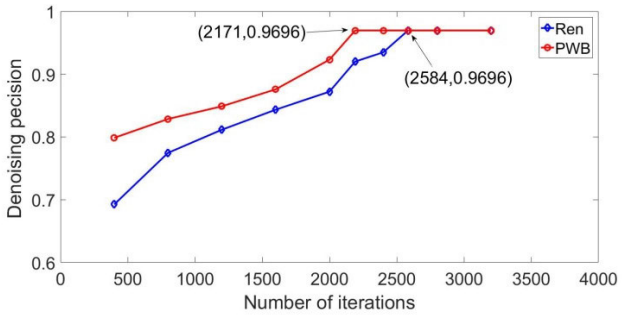



**FIGURE 8.** Here, The red line indicates the de-noising precision curve after PWB processing. The blue line represents the de-noising precision curve processed by the method of Ren.

The de-noising effect is judged in terms of the de-noising precision $\boldsymbol{P_d}$ and the de-noising recall rate $\boldsymbol{R_d}$. The number of noise points removed during the experiment is $\boldsymbol{N_q}$, the point cloud noise to be removed is $\boldsymbol{N_s}$. The total number of deletions is $\boldsymbol{N_y}$, which includes the number of noise points and feature points.

$$P_d = \frac{N_q}{N_s} \tag{13}$$

$$R_d = \frac{N_q}{N_y} \tag{14}$$

From Fig 8 and table 3, at 2584 iterations, the optimal value was obtained by the method of Ren. When iterating 2171 time, the same optimal value was obtained by the method proposed by the study. Regarding the time it takes in order to reach the optimal value, the method proposed by the study is less than that of Ren. It can be seen that the method proposed by the study is superior to Ren's method. The ideal optimization result was obtained by PWB.

**TABLE 3.** The precisions of de-noising are compared by PWB and Ren's method at different iteration times.

| Mothed | The point where the optimal value is reached | Consuming time |
|---|---|---|
| Ren | 2584 | 11.6473s |
| PWB | 2171 | 8.5727s |

In the subsequent experiment, we verify the processing effect of PWB on data sets obtained by the experiment and benchmark data sets respectively. First of all, the experimental dataset is verified by us.

We de-noise the simplified point cloud using five methods, including de-noising of the original threshold, Asokan's optimized bilateral filter method [34], the non-iterative dual-threshold de-noising method [35], Zhang's PCA and bilateral filter method, and the PWB method proposed in this article.

The sphere point cloud is set to M, which is located directly in front of Fig 9. The effect of M's feature retention is used as the standard for desiccation. In Fig 9, According to the experimental data results in Table 3, when the de-noising numbers are 162,354, the speed of PWB is significantly better than that of the other methods. The speed of PWB is 72.31 seconds. The efficiency is 10 times higher than the original threshold denoising method, zhang takes 2.6 times longer than PWB. Therefore, the proposed method has achieved good results on the experimental datasets.

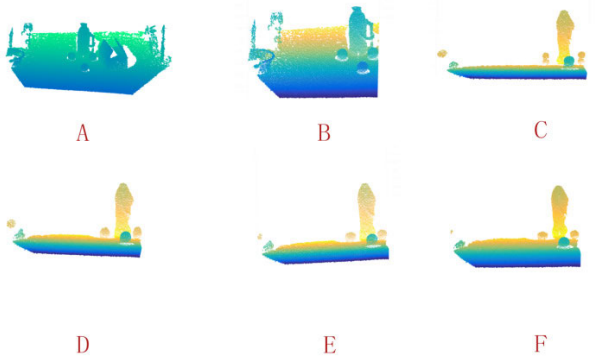



**FIGURE 9.** Here, A represents the original image, B represents the result of original threshold de-noising, C is the result is obtained using Anju Asokan's optimized bilateral filter, D represents the method of Zhang Feng's PCA and bilateral filter, E was obtained by the non-iterative dual-threshold de-noising method, and F is the result obtained by the proposed PWB method.

It can be concluded from Table 4 that the denoising precision of our method is better than Zhang Feng's method.

**TABLE 4.** Comparison of the five denoising methods.

| Algorithm | Total number of noise points | $P_d$ | $R_d$ | Time (s) |
|---|---|---|---|---|
| Original threshold de-noising | 162,354 | 0.3290 | 0.5064 | 759.42 |
| Asokan | 162,354 | 0.5824 | 0.7675 | 248.17 |
| Non-iterative de-noising | 162,354 | 0.7877 | 0.8911 | 221.35 |
| Zhang[25] | 162,354 | 0.9252 | 0.8761 | 189.15 |
| PWB | 162,354 | 0.9696 | 0.9266 | 72.31 |

His method achieved high de-noising accuracy. His method achieved the best results in verification, but our algorithm was even better in the de-noising environment of the model. At the same time, there was a significant effect on the de-noising recall. The result proves that the number of point clouds actually removed is close to the number theoretically removed. Therefore, the removed points are basically noise points. A few feature points are removed, but our method is more accurate than other methods.

On the other hand, it is verified based on the benchmark data set and different parameter conditions. We use three latest artificial neural network methods and PWB to test on the same benchmark data set. Performance of the algorithm is compared with their results of processing. The data set used by Stanford University point cloud scan data sets. The study takes Gaussian noise as an example. Three different coefficients of Gaussian noise are used. Peak Signal to Noise Ratio (PSNR) is invoked as the evaluation standard for the effect of de-noising. In the same experimental image, when the value of PSNR is larger, the processing effect will be better. $\sigma$ is the parameter of Gaussian noise. The results of PSNR are shown in Table 5.

**TABLE 5. The result of the comparison is on the benchmark set.**

| | BRDNet[1] | CTCNN-S[36] | Mothed of Zeng[16] | PWB |
|---|---|---|---|---|
| $\sigma = 1.5$ | 31.10 | 32.47 | 33.15 | 35.28 |
| $\sigma = 2.5$ | 28.97 | 30.29 | 30.81 | 32.54 |
| $\sigma = 5$ | 26.03 | 27.14 | 27.76 | 28.62 |

From Table 5, when $\sigma = 1.5$, the value of PWB is 35.28, it is the best of the four methods. When $\sigma = 2.5$ and $\sigma = 5$, the values are 32.54 and 28.62 respectively. It is also the optimal value among the four methods. Since the value is the highest when $\sigma = 1.5$, the result of processing when $\sigma = 1.5$ is illustrated in in Fig 10.

From Fig 10, the outline of the head of an elephant in the point cloud image was studied as the evaluation object. It can be seen from Fig 10 that the other three methods have the phenomenon of under de-noising and excessive de-noising in the process of de-noising. For example, there are many Gaussian noises scattered around the head of an elephant on BRDNet. For images of CTCNN-S and Zeng, the outline of the head of an elephant as been narrowed relative to the original image, there is excessive de-noising. The PWB-based image retains the outline of the head of an elephant while removing noise. Its result is most similar to the original picture. Therefore, the method proposed in the study has achieved the best results in different Gaussian parameters.

In order to fully demonstrate the effect of the research method, the effect on other data is verified. Fig 11 shows the de-noising results of different Gaussian parameters in the Stanford University point cloud scanning model. The parameter of Gaussian noise $\sigma$ is taken as 2.5, 5, 7.5, 10. Fig 12 is the de-noising results of different Gaussian parameters in the RGB-D dataset. $\sigma$ is taken as 0.1, 0.5, 0.75, 1.

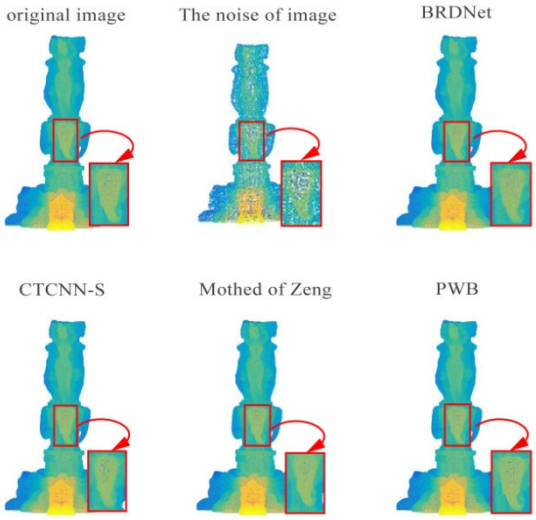



**FIGURE 10. Here, original image is an image that is not disturbed by noise. BRDNet is the image processed by BRDNet. CTCNN-S is the image processed by CTCNN-S. Mothed of Zeng is the image processed by Zeng. PWB is the image processed by PWB.**

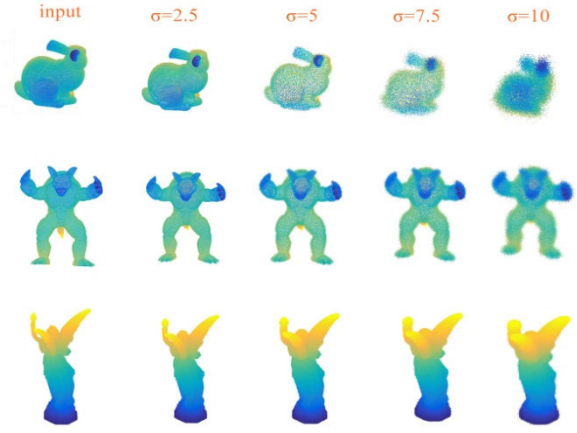



**FIGURE 11. Here, $\sigma = 2.5$ represents the result of point cloud processing by PWB when the Gaussian noise parameter is 2.5. $\sigma = 5$ represents the result of point cloud processing by PWB when the Gaussian noise parameter is 5. $\sigma = 7.5$ represents the result of point cloud processing by PWB when the Gaussian noise parameter is 7.5. $\sigma = 10$ represents the result of point cloud processing by PWB when the Gaussian noise parameter is 10.**

From Fig 11, it can be seen that our method has an excellent effect when $\sigma$ is less than 7.5, especially when it is less than 2.5. When $\sigma$ is more important than 7.5, the effect of de-noising will be lacking. It can be seen from Fig 12 that the method can have a good effect when it is lower than 0.5, especially when it is 0.1. When it is larger than 0.5, the effect is not good.

At last, the effect of the method has been confirmed by us in additional fields. Parameters in other fields are changing. SET5, SET14 and BSDS100 datasets are selected. RSNR, (structural similarity index) SSIM and IFC are used as evaluation criteria, where SSIM is the contrast of the structure of the final image and the original image. When the value of SSIM

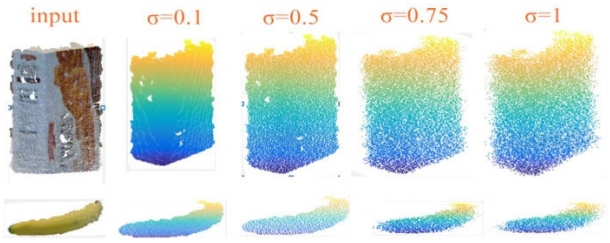



**FIGURE 12.** Here, $\sigma = 0.1$ represents the result of point cloud processing by PWB when the Gaussian noise parameter is 0.1. $\sigma = 0.5$ represents the result of point cloud processing by PWB when the Gaussian noise parameter is 0.5. $\sigma = 0.75$ represents the result of point cloud processing by PWB when the Gaussian noise parameter is 0.75. $\sigma = 1$ represents the result of point cloud processing by PWB when the Gaussian noise parameter is 1.

is larger, it is proved that the similarity is high and the effect is better. The results are presented in Table 6 and Table 7. The optimal value of each item is displayed in bold.

**TABLE 6.** Results based on SET5 and SET14 data sets.

| Alorithm | SET5 | | | SET14 | | |
|---|---|---|---|---|---|---|
| | RSNR | SSIM | IFC | PSNR | SSIM | IFC |
| VDSR[37] | 37.53 | 0.958 | 8.190 | 32.97 | 0.913 | 7.787 |
| DRCN[38] | 37.63 | 0.959 | 8.326 | 32.98 | 0.913 | 8.025 |
| LAPSRN[39] | 37.52 | 0.959 | 9.010 | **33.08** | 0.913 | 8.505 |
| PWB | **37.81** | **0.967** | **9.081** | 32.99 | **0.916** | **8.542** |

**TABLE 7.** Results based on BSDS100 data sets.

| Alorithm | BSDS100 | | |
|---|---|---|---|
| | RSNR | SSIM | IFC |
| VDSR | 31.90 | 0.896 | 7.169 |
| DRCN | 31.85 | 0.894 | 7.220 |
| LAPSRN | 31.80 | 0.895 | 7.715 |
| PWB | **32.54** | **0.899** | **7.938** |

The comparison of PSNR, SSIM and IFC for each method is shown in Fig 13, Fig 14 and Fig 15.

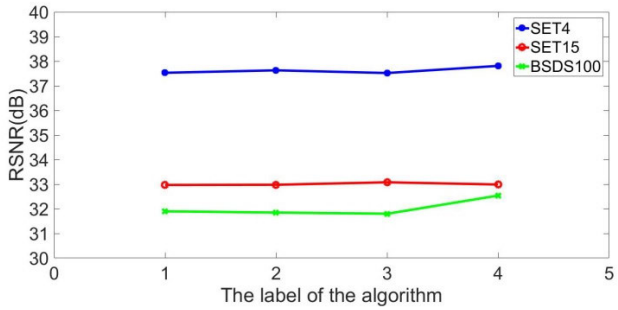



**FIGURE 13.** Here, 1 represents the result of VDSR. 2 represents the result of DRCN. 3 represents the result of LAPSRN. 4 represents the result of PWB.

From Table 6, the proposed method was verified on the SET5 data set, the optimal values were obtained by all three

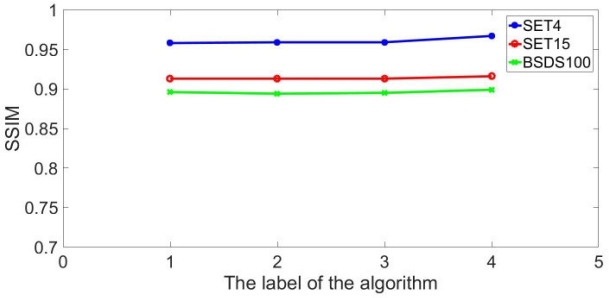



**FIGURE 14.** Here, 1 represents the result of VDSR. 2 represents the result of DRCN. 3 represents the result of LAPSRN. 4 represents the result of PWB.

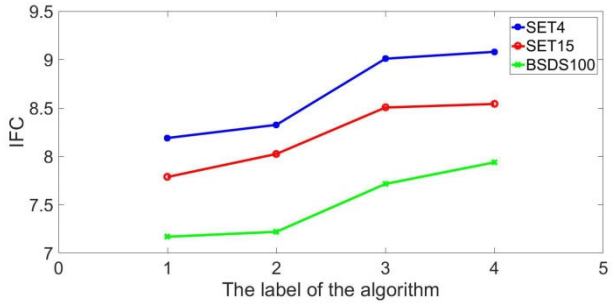



**FIGURE 15.** Here, 1 represents the result of VDSR. 2 represents the result of DRCN. 3 represents the result of LAPSRN. 4 represents the result of PWB.

evaluation criteria. These results prove that PWB is better than the other three methods based on artificial neural network. From Table 6, Fig 13 and Fig 15, the optimal values of PNSR are obtained by LAPSRN. These optimal values of SSIM and IFC are obtained by PWB. The PNSR of PWB is 32.99. The PNSR of LAPSRN is 33.08. The two values are different from 0.09. PWB can be further optimized in future work. Better results will be achieved. From Table 7, PNSR, SSIM, and IFC, PWB achieved results of 32.54, 0.899, and 7.939, these results are superior to the other three methods.

From comparing different parameters, different datasets, and different methods, the ideal results are obtained by FWD and PWB in diverse environments.

## IV. CONCLUSION

In the experiment, the unevenness of a spherical point cloud is difficult to simplify. When it makes a 3D voxel grid, the traditional under-sampling process has the limitation of selecting the center of gravity as the feature point, so some key points will be lost. We proposed a point cloud reduction method that uses the farthest-sampled points combined with the weighted average method to optimize down-sampling, namely, the FWD method. The center of gravity is found by building a voxel grid. The ten points are sequentially found by using the furthest-point sampling principle in the voxel grid. The final point is calculated by the weighted average of the distances of ten points. The resulting points are taken as characteristic points. Experiments prove that the problems

caused by unevenness are solved. This method is versatile and can be applied to plane and uneven surfaces, allowing point cloud data to be effectively simplified. Our method is also robust. The shapes of objects are well preserved and object deformation is prevented. The proposed method has achieved excellent results on both experimental data sets and benchmark data sets. On the experimental data set, we simplified 90840 points in 48 seconds and kept the feature points intact. Compared with other latest methods, better results have been obtained. Experiments prove the feasibility of FWD.

Secondly, the influence of external points of the target on processing is considered. For example, the extraction of feature points is affected by marginal isolated points. It is easy to produce Gaussian noise. The Gaussian function is used in the existing bilateral filtering process. The influence of pixels on the domain is ignored. A problem of insufficient denoising is caused. The PWB method proposed by us is based on a combination of optimized PCA and bilateral filtering. For the optimization of bilateral filtering, the parameters are optimized with PSO by us, through comparison with Ren, the results prove that PWB can achieve de-noising precision faster. The Gaussian function is replaced by wavelet filtering in the selection of smoothing parameters. The problem of insufficient de-noising is caused by pixels in the domain that are not considered. This problem noising is overcome by our method and the effects of marginal outliers are better resolved. Experiments show that de-noising of point clouds and protection of feature points can be better achieved by our method. In the experimental set, de-noising precision of 0.9696 was completed by us. PWB is 4 percentage points higher than the latest method. On the benchmark datasets, the optimal PSNR value is obtained by PWB under different Gaussian noise, when $\sigma = 1.5$, the PSNR value obtained by PWB is 35.28, PWB is 2% higher than the latest method based on artificial neural network. There is also verification of applications in other fields. PWB has achieved excellent results. Therefore, the performance of PWB proved to be excellent.

Not all point clouds can be fully involved in the wavelet transform process proposed in this article. Some of the point clouds do not satisfy the wavelet transform conditions, causing distortion of the point cloud. A solution to this is urgently needed and is a direction for future research.

## REFERENCES


[1] C. Tian, Y. Xu, and W. Zuo, "Image denoising using deep CNN with batch renormalization," *Neural Netw.*, vol. 121, pp. 461–473, Jan. 2020.

[2] Y.-H. Shin, M.-J. Park, O.-Y. Lee, and J.-O. Kim, "Deep orthogonal transform feature for image denoising," *IEEE Access*, vol. 8, pp. 66898–66909, 2020.

[3] W. Shi, F. Jiang, S. Zhang, R. Wang, D. Zhao, and H. Zhou, "Hierarchical residual learning for image denoising," *Signal Process., Image Commun.*, vol. 76, pp. 243–251, Aug. 2019.

[4] J. Lu, J. Xuan, G. Zhang, and X. Luo, "Structural property-aware multilayer network embedding for latent factor analysis," *Pattern Recognit.*, vol. 76, pp. 228–241, Apr. 2018.

[5] Y. Wang, Y. Cao, Z. Guo, and S. Wen, "Passivity and passification of memristive recurrent neural networks with multi-proportional delays and impulse," *Appl. Math. Comput.*, vol. 369, Mar. 2020, Art. no. 124838.

[6] B. Sun, Y. Cao, Z. Guo, Z. Yan, and S. Wen, "Synchronization of discrete-time recurrent neural networks with time-varying delays via quantized sliding mode control," *Appl. Math. Comput.*, vol. 375, Jun. 2020, Art. no. 125093.

[7] B. Sun, S. Wen, S. Wang, T. Huang, Y. Chen, and P. Li, "Quantized synchronization of memristive neural networks with time-varying delays via super-twisting algorithm," *Neurocomputing*, vol. 380, pp. 133–140, Mar. 2020.

[8] F. Gu, H. Zhang, and C. Wang, "A two-component deep learning network for SAR image denoising," *IEEE Access*, vol. 8, pp. 17792–17803, 2020.

[9] I. Hong, Y. Hwang, and D. Kim, "Efficient deep learning of image denoising using patch complexity local divide and deep conquer," *Pattern Recognit.*, vol. 96, Dec. 2019, Art. no. 106945.

[10] S. K. Ghosh, B. Biswas, and A. Ghosh, "SDCA: A novel stack deep convolutional autoencoder—An application on retinal image denoising," *IET Image Process.*, vol. 13, no. 14, pp. 2778–2789, Dec. 2019.

[11] J. H. Choi, O. A. Elgendy, and S. H. Chan, "Optimal combination of image denoisers," *IEEE Trans. Image Process.*, vol. 28, no. 8, pp. 4016–4031, Aug. 2019.

[12] R. S. Thakur, R. N. Yadav, and L. Gupta, "State-of-art analysis of image denoising methods using convolutional neural networks," *IET Image Process.*, vol. 13, no. 13, pp. 2367–2380, Nov. 2019.

[13] M. Elhoseny and K. Shankar, "Optimal bilateral filter and convolutional neural network based denoising method of medical image measurements," *Measurement*, vol. 143, pp. 125–135, Sep. 2019.

[14] N. Li, S. Yue, and B. Jiang, "Adaptive and feature-preserving bilateral filters for three-dimensional models," *Traitement Du Signal*, vol. 37, no. 2, pp. 157–168, Apr. 2020.

[15] C. Dinesh, G. Cheung, and I. V. Bajic, "3D point cloud color denoising using convex graph-signal smoothness priors," in *Proc. IEEE 21st Int. Workshop Multimedia Signal Process. (MMSP)*, Sep. 2019, pp. 1–6. [Online]. Available: https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=GeneralSearch&qid=9&SID=8BlkDoNwF9VBoxG5Y7r&page=1&doc=1

[16] J. Zeng, C. Gene, M. Ng, J. Pang, and C. Yang, "3D point cloud denoising using graph Laplacian regularization of a low dimensional manifold model," *IEEE Trans. Image Process.*, vol. 29, pp. 3474–3489, 2020.

[17] S. Birogul, G. Temur, and U. Kose, "YOLO object recognition algorithm and 'buy-sell decision' model over 2D candlestick charts," *IEEE Access*, vol. 8, pp. 91894–91915, 2020.

[18] J. Qi, W. Hu, and Z. Guo, "Feature preserving and uniformity-controllable point cloud simplification on graph," in *Proc. IEEE Int. Conf. Multimedia Expo (ICME)*, Jul. 2019, pp. 284–289. [Online]. Available: https://apps.webofknowledge.com/Search.do?product=WOS&SID=8BlkDoNwF9VBoxG5Y7r&search_mode=GeneralSearch&prID=f63158a9-c320-4d3d-b695-35fa79d27280

[19] V. Markovic, Z. Jakovljevic, and Z. Miljkovic, "Feature sensitive three-dimensional point cloud simplification using support vector regression," *Tehni ki Vjesnik-Tech. Gazette*, vol. 26, no. 4, pp. 985–994, Jul. 2019.

[20] Y. Li, H. Liu, Y. Tao, and J. Liao, "Reasoning mechanism: An effective data reduction algorithm for on-line point cloud selective sampling of sculptured surfaces," *Comput.-Aided Des.*, vol. 113, pp. 48–61, Aug. 2019.

[21] C. Ji, Y. Li, J. Fan, and S. Lan, "A novel simplification method for 3D geometric point cloud based on the importance of point," *IEEE Access*, vol. 7, pp. 129029–129042, 2019.

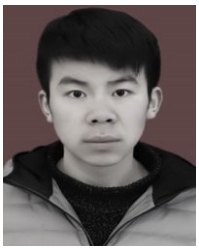
[22] P. Hermosilla, T. Ritschel, and P. P. Vazquez, "Monte Carlo convolution for learning on non-uniformly sampled point clouds," in *Proc. 11th ACM SIGGRAPH Conf. Exhib. Comput. Graph. Interact. Techn. Asia (SA)*, 2018, [Online]. Available: https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=GeneralSearch&qid=35&SID=5EgYXNtyMYTSFjt5BZX&page=1&doc=1

[23] Q. Du, "3D point cloud registration denoising method for human motion image using deep learning algorithm," *Multimedia Syst.*, vol. 26, no. 1, pp. 75–82, Feb. 2020.

[24] Z. Yang and D. Xiao, "A systemic point-cloud de-noising and smoothing method for 3D shape reuse," in *Proc. 12th Int. Conf. Control Autom. Robot. Vis. (ICARCV)*, Dec. 2012, pp. 1722–1727. [Online]. Available: https://apps.webofknowledge.com/Search.do?product=WOS&SID=8BlkDoNwF9VBoxG5Y7r&search_mode=GeneralSearch&prID=d2122c41-ee15-4601-9159-265e3951c691

[25] F. Zhang, C. Zhang, H. Yang, and L. Zhao, "Point cloud denoising with principal component analysis and a novel bilateral filter," *Traitement Du Signal*, vol. 36, no. 5, pp. 393–398, Nov. 2019.

[26] G. Ren, Y. Cao, S. Wen, T. Huang, and Z. Zeng, "A modified elman neural network with a new learning rate scheme," *Neurocomputing*, vol. 286, pp. 11–18, Apr. 2018.
[27] W.-J. Niu, Z.-K. Feng, Y.-B. Chen, H.-R. Zhang, and C.-T. Cheng, "Annual streamflow time series prediction using extreme learning machine based on gravitational search algorithm and variational mode decomposition," *J. Hydrolog. Eng.*, vol. 25, no. 5, May 2020, Art. no. 04020008.
[28] Z.-K. Feng, W.-J. Niu, R. Zhang, S. Wang, and C.-T. Cheng, "Operation rule derivation of hydropower reservoir by k-means clustering method and extreme learning machine based on particle swarm optimization," *J. Hydrol.*, vol. 576, pp. 229–238, Sep. 2019.
[29] Z. R. Hao, C. Li, and W. L. Ren, "Optimization design of water jet propulsion pump blade based on improved PSO algorithm," *J. Drainage Irrigation Machinery Eng.*, vol. 38, no. 6, pp. 566–570, 2020.
[30] W. L. Ren, Z. R. Hao, and Y. Wang, "Application of improved partical swarm algorithm in 3D design of hydrofoil," *J. Jiangsu Univ.*, vol. 38, no. 2, pp. 168–172, 2017.
[31] B. Ren and H. Li, "A wavelet filter algorithm for digital image," *J. Shenyang Inst. Technol.*, vol. 1, pp. 47–49, 2004.
[32] Zhu, Kukko, Virtanen, Hyyppä, Kaartinen, Hyyppä, and Turppa, "Multisource point clouds, point simplification and surface reconstruction," *Remote Sens.*, vol. 11, no. 22, p. 2659, Nov. 2019.
[33] M. Shoaib, J. Cheong, Y. Kim, and H. Cho, "Fractal bubble algorithm for simplification of 3D point cloud data," *J. Intell. Fuzzy Syst.*, vol. 37, no. 6, pp. 7815–7830, Dec. 2019.
[34] A. Asokan and J. Anitha, "Adaptive cuckoo search based optimal bilateral filtering for denoising of satellite images," *ISA Trans.*, vol. 100, pp. 308–321, May 2020.
[35] S. Zhou, X. Liu, C. Wang, and B. Yang, "Non-iterative denoising algorithm based on a dual threshold for a 3D point cloud," *Opt. Lasers Eng.*, vol. 126, Mar. 2020, Art. no. 105921.
[36] Z. Lyu, C. Zhang, and M. Han, "A nonsubsampled countourlet transform based CNN for real image denoising," *Signal Process., Image Commun.*, vol. 82, Mar. 2020, Art. no. 115727.
[37] J. Kim, J. K. Lee, and K. M. Lee, "Accurate image super-resolution using very deep convolutional networks," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1646–1654. [Online]. Available: https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=GeneralSearch&qid=21&SID=5EgYXNtyMYTSFjt5BZX&page=1&doc=7
[38] J. Kim, J. K. Lee, and K. M. Lee, "Deeply-recursive convolutional network for image super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jun. 2016, pp. 1637–1645. [Online]. Available: https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=GeneralSearch&qid=25&SID=5EgYXNtyMYTSFjt5BZX&page=1&doc=2
[39] W.-S. Lai, J.-B. Huang, N. Ahuja, and M.-H. Yang, "Deep Laplacian pyramid networks for fast and accurate super-resolution," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 624–632. [Online]. Available: https://apps.webofknowledge.com/full_record.do?product=WOS&search_mode=GeneralSearch&qid=29&SID=5EgYXNtyMYTSFjt5BZX&page=1&doc=2



**BOCHANG ZOU** received the B.E. degree from the Qilu University of Technology, Jinan, China, in 2018, where he is currently pursuing the master's degree. His research interests include image processing, machine vision, 3D point cloud processing, and object recognition.

**HUADONG QIU** received the B.E. degree from the Chengdu University of Technology, in 2007, and the Ph.D. degree from Shandong University, Jinan, China, in 2012. Excellence Master Thesis honor the thesis was included in the Chinese Selected Doctoral Dissertations and Master's Theses Full-Text Databases.

He was a Researcher with the National Institute of Testing Technology, China, from 2005 to 2007. He has participated in the research project from the National Institute of Metrology, China, and the National Institute of Testing Technology, China, as a Key Researcher. His research project is numerical transferable/ traceability system of environmental radioactivity. He evaluated the ORTEC's HPGe gamma-ray spectroscopy stability performance and calibrated the apparatus's efficiency and energy. He finished the master's thesis about radioactive statistics process control and received Outstanding Master's Thesis honor. His research interests include image processing, machine vision, deep learning, neural network learning, and artificial intelligence.

**YUFENG LU** received the Ph.D. degree from Shandong University, Jinan, China, in 2008.

He has been an Associate Professor with the School of Mechanical and Automotive Engineering, Qilu University of Technology, since 2008. He was a Visiting Scholar with the Department of Automotive Engineering, Tsinghua University, from September 2015 to July 2016. He undertook the key research project of Shandong Province Research and development of self-driving trailer with braking energy recovery. He has published more than ten articles, one monograph, three authorized patents, and four software copyrights. His research interests include intelligent vehicle and vehicle energy saving.